## REMARKS

1. Claims 14-18, 20-29 were rejected under 35 U.S.C. 112, second paragraph.

Claim 14 is amended to improve clarity per the Examiner's comments.

Regarding Claims 15 and 24, the Examiner states:

> Claims 15, and 24 is rejected ... as being incomplete for omitting essential structural cooperative relationships of elements, such omission amounting to a gap between the necessary structural connections. See MPEP § 2172.01. The omitted ... relationships are: the step between suspending the first instruction and executing the first instruction. It is unclear as to what steps the processor is taking in regard to first instruction to go to other instructions for processing, ... and ... what step does the processor takes to go back to the first instruction.

MPEP 2701.02 states:

> A claim which omits matter **disclosed to be essential** to the invention **as described in the specification or in other statements of record** may be rejected under 35 U.S.C. 112, first paragraph, as not enabling. ... Such essential matter may include missing elements, steps or necessary structural cooperative relationships of elements **described by the applicant(s) as necessary** to practice the invention.

The specification describes certain techniques for the processor to suspend the first instruction and go to other instructions and then to go back to the first instruction (see e.g. Figs. 6-13B), but the specification does not indicate that these or other techniques are essential or necessary. Applicants are allowed to draft the claims broadly without limiting the claims to such techniques. See MPEP 2173.04 ("Breadth Is Not Indefiniteness").

Claim 18 is amended to overcome the rejection ("TA1" is replaced with --TA2--).

Regarding Claims 19 and 26, the Examiner states:

> ... claims 19, and 26 ... is rejected under 35 U.S.C. 112, second paragraph, as being incomplete for omitting essential elements, such omission amounting to a gap between the elements. See MPEP § 2172.02. The omitted elements are: the relationship between the ... first circuitry ...

> and the ... second circuitry... The claim is so vague that it does not permit understanding of the limitations ...

Claim 19 is supported by Fig. 13B. This figure shows four circuits 1390 only one of which is illustrated in detail (specification, page 38, lines 15-18). Each circuit 1390 includes a task next state generator 1398. The "first signal" of Claim 19 reads on the suspend signals on lines 410 provided by circuit 1401 to generators 1398 (only one of the suspend signals is shown in Fig. 13B). See specification, page 39, lines 1-4. Generators 1398 are responsive to the suspend signals to generate the signals A "indicating whether the task remains suspended or, alternatively, whether the task can be made active ..." (page 39, lines 5-7). Arbiter 1403 receives the A signals and schedules the tasks for execution by generating the active software task numbers (page 39, lines 12-16). Thus, the task scheduling is performed in response to the suspend signals as recited in Claim 19.

Claim 19 is not limited to the embodiment of Fig. 13B.

MPEP 2172.01 does not apply because the specification does not state that any particular implementation of Claim 19 is essential or necessary. See the discussion above in connection with Claims 15 and 24.

Claim 26 is believed to be allowable for similar reasons.

Regarding Claim 22, the Examiner states:

> ... it is unclear as to which task's is being scheduled while the second circuitry is scheduling the instruction. Its own task or another tasks instruction.

Claim 22 is supported by Fig. 6, showing five instructions 1-5. Each instruction is executed in pipeline stages t0-t6. The stage t0 is a task scheduling stage TS. "In this stage, an active task is selected", i.e. a task is selected that will execute the instruction (page 22, lines 3-4). The instruction to be executed is then determined based on the program counter register (PC register) corresponding to the selected task. There are sixteen PC registers, one for each task (specification, page 21, lines 19-20). The PC register corresponding to the

selected task is used to determine the instruction that will be executed in stages t1-t6. Specification, page 21, lines 18-20, and page 22, lines 26-30.

Claim 22 is not limited to the embodiments discussed herein.

Regarding Claim 25, the language objected to by the Examiner is deleted.

Claim 29 has a language similar to the language of Claim 22, and is supported by the original disclosure as discussed above. Claim 29 is not limited to the embodiments discussed herein.

2. Claims 14-29 were rejected under 35 U.S.C. 102 over Okin. "Okin" is assumed to be U.S. patent no. 5,361,337, as in the Notice of References Cited. (There seems to be an error in the patent number on page 5 of the Office Action.)

**Claim 14 and its dependent Claims 15, 17,19-22, 30-49**

**Claim 14** is supported by Applicants' Figs. 5 and 6 and the specification page 23. An instruction can be blocked in pipeline stage t3 (Read) by a Suspend signal (page 23, lines 7-8) or a Wait signal (page 23, lines 15-16). If the instruction is blocked, the instruction is later re-executed (page 23, lines 13-14 and 19-23).

Claim 14 is not limited to the embodiments discussed herein.

Okin teaches duplicating the state flip-flops of a processor to allow an instruction execution pipeline to "switch from one process to another" if a cache miss occurs (column 3, lines 34-35, column 4, lines 1-2). "Having multiple copies of state elements on the processor ... permits the processor to save the context of the current instructions and resume executing new instructions" (Abstract). Okin does not teach or suggest that the processor later re-executes the current instruction (and not simply resumes the current instruction) when the data are brought into the cache. Therefore, Okin does not teach or suggest Applicants' invention.

In some embodiments, re-execution of an instruction is simpler and does not require duplication of flip-flops needed to keep the state of the current instruction. (Claim 14 is not limited to such embodiments however.)

**Claim 22** further recites scheduling a task or tasks for execution on each instruction.

Okin does not teach or suggest scheduling a task or tasks on each instruction in the absence of a cache miss.

**Claim 30** recites that "the processor again fetches the first instruction from the memory to re-execute the first instruction." Claim 30 is supported by Fig. 6 and specification, page 22, lines 14 and 28-30 (fetching the instruction from memory 314 in the F pipeline stage in each instruction execution).

Claim 30 is not limited to the embodiments discussed herein.

Okin does not teach or suggest fetching the instruction again on a cache miss. To the extent that Okin's duplication of the state elements may avoid the need to re-fetch the instruction, Okin teaches away from the invention of Claim 30.

**Claim 31** recites that "the first instruction is aborted after being decoded ..., and the first instruction is decoded again when the first instruction is being re-executed". This recitation is supported by Applicants' Fig. 6 and page 23. The instruction is decoded in the D stage t2 (page 23, lines 1-2), and aborted based on the Suspend or Wait signal generated in the next stage t3 (page 23, lines 3-23). If instruction no. 1 is aborted, it may be re-executed as instruction no. 5 (page 23, lines 22-23). The decode stage will be performed to again decode the instruction.

Claim 31 is not limited to the embodiments described herein.

Okin does not teach or suggest decoding the instruction again on a cache miss. To the extent that Okin's duplication of the state elements may avoid the need to re-decode the instruction, Okin teaches away from the invention of Claim 31.

**Claims 43, 44** are believed to be allowable for similar reasons.

**Claims 18, 23-29, 50-68**

Claim 18 recites a resource "unavailable to the task TA1 due to the resource being made available to another task". This recitation is supported by Fig. 4 and specification, page 19, lines 14-18 (Task 0 is not allowed to read the request FIFO when the FIFO is made available to Task 1). Claim 18 is not limited to the embodiments discussed herein.

Okin's cache misses are unrelated to the cache being made available to another process or task as recited in Claim 18.

**Claim 23** recites re-executing an instruction. See the discussion above of Claim 14.

**Claims 24-29, 50-68** depend from Claim 23. In addition:

- **Claim 29** is believed to be allowable for reasons similar to the reasons given above for Claim 22.

- **Claim 50** is believed to be allowable for reasons similar to the reasons given above for Claim 30.

- **Claims 51, 62, 63** are believed to be allowable for reasons similar to the reasons given above for Claims 31, 43, 44.

Any questions regarding this case can be addressed to the undersigned at the telephone number below.

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on April 13, 2004.

Michael Shenker 4-13-04
Attorney for Applicant(s) Date of Signature

Respectfully submitted,

Michael Shenker

Michael Shenker
Patent Attorney
Reg. No. 34,250
Telephone: (408) 392-9250

Law Offices Of
MacPherson Kwok Chen & Heid LLP
1762 Technology Drive, Suite 226
San Jose, CA 95110